STOCHASTIC PULSE CODED ARITHMETIC

S.L. Toral, J.M. Quero, Member, IEEE and L.G. Franquelo, Senior Member, IEEE

Department of Electronic Engineering, University of Seville
Avda. Camino de los Descubrimientos, 41092, Seville, Spain

ABSTRACT

Among the different pulse codification techniques, stochastic pulse codification has its own arithmetic based on the similarity between boolean algebra and statistic algebra. Summation and multiplication are the two basic arithmetic operations deeply treated in literature. In this paper we present two digital stochastic circuits that extend traditional stochastic algebra: a division circuit and a square-root circuit, and the interfaces between the analog and stochastic domain.

As result, we are able to process analog input signals with a simple and complete processing system. These circuits can be implemented in low-cost and low-power digital programmable devices.

1. INTRODUCTION

Stochastic systems make pseudo analog operations using stochastically coded pulse sequences [1], [2]. Information is represented by the statistical mean value of a pulse sequence. In a binary logic, it is the probability of taking a "high" level. Figure 1 shows the generation of a stochastic pulse stream from a digital value. The value stored

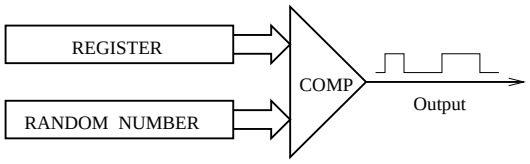

Figure 1: Digital to stochastic conversion (DSC).

in a register is compared with a random number generator. If the random number generator is less or equal than the register value, the output of the comparator is set to a high level. Otherwise, a low level is set. Figure 1 can be considered as the scheme of a *digital to stochastic converter*. Equation 1 give us the probability of an output high level.

$$P(Output ='' 1'') = \frac{Digital\ Value}{2^n} \qquad (1)$$

The stochastic pulse stream at the output of the comparator is a Bernouilli's sequence, with probability of a "high" level output equal to the number stored in the register. Bernouilli's theorem guarantees that the mean of the addition of the Bernouilli's random variables tends to that probability. So, the more pulses we process, the best accuracy we obtain when representing a digital value by means of a stochastic pulse stream.

There are several methods to generate random numbers. Digitally, we can implement a pseudo random number generator, that is, a sequence of number with randomness appearance, but exhibiting a repeatable pattern. Among the whole variety of pseudo random number generators [3], the most extended one is the linear feedback shift register (LFSR) of maximum length [4].

Stochastic pulse coded arithmetic is very close to processing with oversampled signals [5]. The main difference is the following: while pulse density arithmetic only performs basic operations such us addition and multiplication, stochastic pulse coded arithmetic can perform supplementary operations that will be presented in this paper. This approach allow a very simple electronic implementation of real applications. Particularly, a reactive power measurement method and a hardware electronic implementation of a passivity based controller for a series resonant converter have been proposed in [6] and [7].

In next section, we will explain basic conversion schemes between analog and stochastic domain. Section III describes the circuits that perform traditional stochastic computation, that is, summation and multiplication. Then, division and root-square circuits will be presented. Finally, conclusions and applications of these pulse coded processing system will be pointed out.

2. CONVERSION SCHEMES

A digital or analog value can be recovered from a digital stochastic pulse stream by estimating its mean value in a long enough sequence. A counter can perform the estimation in the digital domain and a RC circuit is suitable for the analog one.

Statistically, mean value estimator has a Gaussian distribution with a mean value equals to the mean value of the discrete Bernouilli sequence and a standard deviation that decreases with root-square of the length N of that sequence (equation 2)

$$mean\ \ value\ \ estimator : AN(\mu, \frac{\sigma}{\sqrt{N}}) \qquad (2)$$

Figure 2 illustrates an analog signal extracted from a stochastic pulse stream generated with the general scheme LFSR plus a comparator. A complete study of this circuit can be found in [8]. Analog to stochastic conversion is a more complicated operation to perform. All the

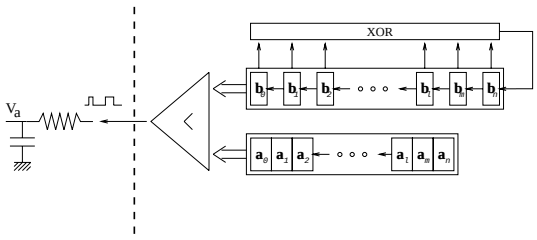

Figure 2: Analog mean value of a stochastic pulse stream

circuits proposed are based in a negative feedback scheme in which the analog input is compared with the analog signal obtained from the digital stochastic pulse stream [9]. The problem is that the RC integration of the digital pulse stream of figure 3 is not a fast integration, because cut off frequency of the RC filter must be low to recover the mean value. So, the bandwidth of the converter is limited below 1kHz for 8-bit accuracy and a digital clock frequency of 10 MHz [9]. In [10] has been proposed a novel

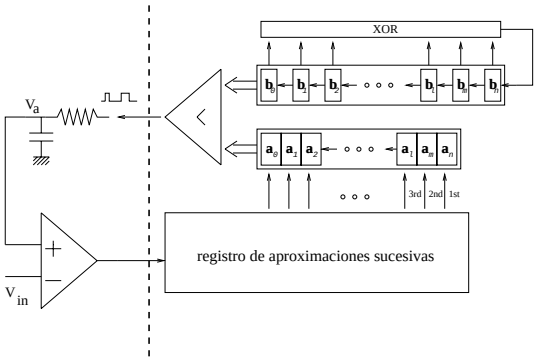

Figure 3: Analog to stochastic conversion

conversion circuit based on sigma-delta modulation that improve bandwidth and accuracy. The reason is that, in figure 4, we integrate the error signal between the analog input and the stochastic pulse stream, without a previous RC filtering. Once we have presented the converters be-

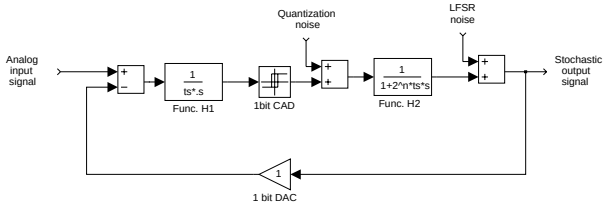

Figure 4: Analog to stochastic conversion based on sigma-delta modulation

tween analog and digital domain and stochastic domain, it is possible to define a stochastic processing system in which, given a set of analog or digital inputs, they are stochastically converted, then processed, and finally recovered from the stochastic pulse stream as an analog or digital value. Figure 5 is a block diagram that illustrates the whole processing system. The main advantage of the stochastic processing system is the possibility of doing pseudo-analog functions working with the mean value of the pulse stream, but with a digital implementation.

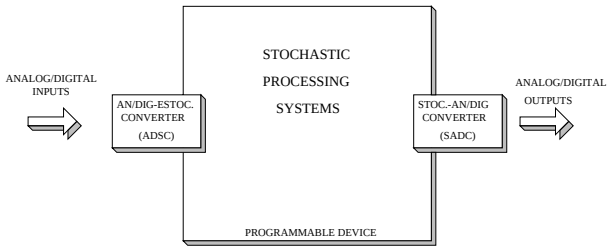

Figure 5: Stochastic processing system

3. STOCHASTIC ARITHMETIC

3.1. Basic operations

Traditional operations defined in the literature are product and summation [11]. This is because they are the typical operations involved en each arithmetic computation and they are also extensively used in neural networks, which is one of the applications of stochastic logic.

Particularly, the product is performed by an AND gate, provided that the input stochastic pulse streams are uncorrelated. If p_1 and p_2 are the input pulse streams of the AND gate, the output pulse stream will only have a high level when both p_1 and p_2 have a high level. Then the output of the AND gate has a probability equals to the product of the input probabilities.

Summation is a more complicated operation to perform. The reason is that, using an OR gate, the operation carried out is not actually addition but $p_1 + p_2 - p_1 \cdot p_2$. As an adder, the OR gate has the disadvantage of saturation when a pulse overlapping occurs. The OR gate is well suited when we are working with low pulse densities. In the circuits proposed, it will be used a signed adder scheme, based on the truth table 1. It is a combinational block, with the following logic equations (3):

$$\begin{aligned} add(p_1, p_2) &= p_1 \oplus p_2 + p_1 \overline{(sig(p_1) \oplus sig(p_2)} \\ sig[add(p_1, p_2)] &= p_1 sig(p_1) + p_2 sig(p_2) \end{aligned} \tag{3}$$

Finally, integration is performed by an up/down

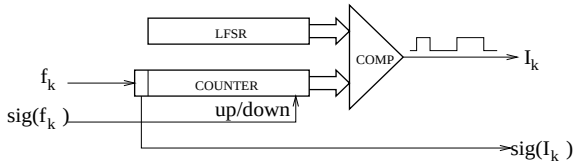

Figure 6: Stochastic integrating circuit

$ADDER$ block					
p_1	sp_1	p_2	sp_2	(p_1+p_2)	$s(p_1+p_2)$
0	0	0	0	0	-
0	0	0	1	0	-
0	0	1	0	1	0
0	0	1	1	1	1
0	1	0	0	0	-
0	1	0	1	0	-
0	1	1	0	1	0
0	1	1	1	1	1
1	0	0	0	1	0
1	0	0	1	1	0
1	0	1	0	1	0
1	0	1	1	0	-
1	1	0	0	1	1
1	1	0	1	1	1
1	1	1	0	0	-
1	1	1	1	1	1

Table 1: Truth table of $ADDER$ block

counter that stores in time the stochastic pulse stream $(f_k, sig(f_k))$. By comparing the counter with a LFSR (figure 6), we have an output pulse stream representing input signal integration $(I_k, sig(I_k))$ [12].

3.2. Supplementary operations

Based on circuits for product and summation, we will extend traditional basic stochastic operations by means of a division circuit and a square root circuit.

Division circuit is showed in figure 7. It is a negative feedback scheme in which we are looking for a stochastic pulse stream p_1/p_2 that, multiplied by p_2, is compared with the input pulse stream, increasing or decreasing the counter according to the error pulse stream $e_k, sig(e_k)$. As

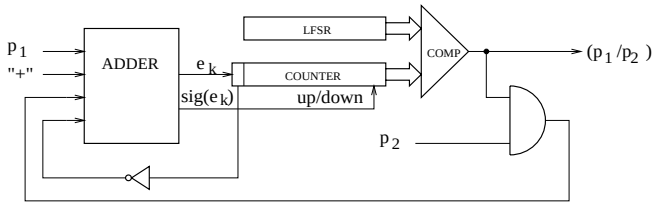

Figure 7: Stochastic division circuit

we are using a negative feedback, the error signal tends to zero. So, the use of the adder block is justified because we are working with low pulse densities. As we can only multiply by a value in the range [0,1] (the value codified is a probability), it must be verified that $p_1 \leq p_2$.

Experimental results has been obtained using a digital programmable device, with the converters of section II, a counter size of 10 bits and a digital clock frequency of 12 MHz. Figure 8 shows experimental results as a function of p_2 for different values of p_1, and compared to the theoretical result.

Square root circuit has a similar (figure 9) scheme but using the possibilities of redundancy in information of

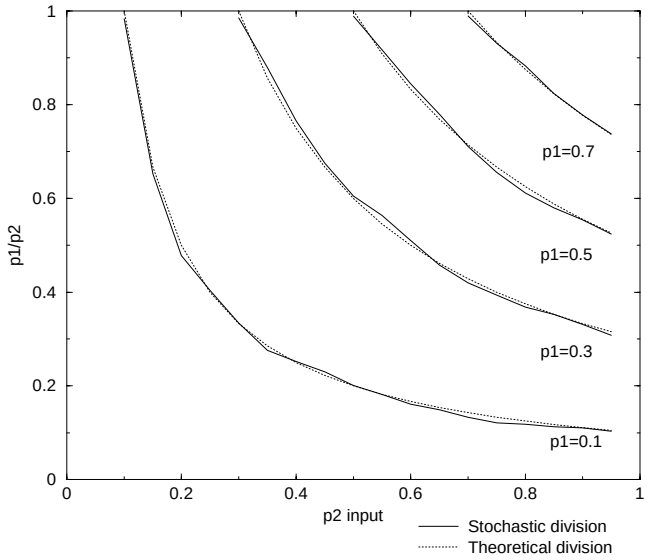

Figure 8: Experimental results of the stochastic division circuit as a function of p_2 for different values of p_1

stochastic logic. Two different pulse stream can represent the same value with a different pattern of pulses. We are looking for a stochastic pulse stream that, multiplied by itself using an uncorrelated version, tends to the input pulse stream. The counter is increased or decreased according to the error signal, the same than the previous circuit. The pulse streams y_1 and y_2 are redundant and both of them represents the square root of the input pulse stream p_1. Experimental results has been obtained us-

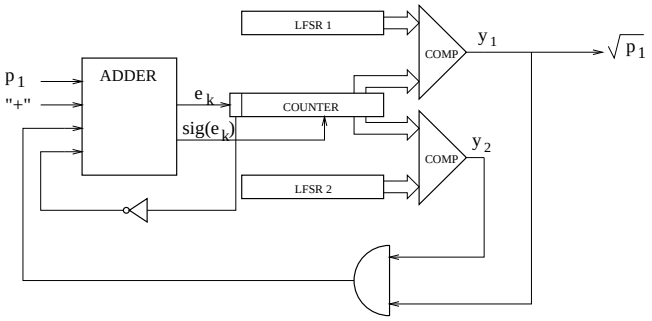

Figure 9: Stochastic square root circuit

ing the converter of section II, leading to the figure 10 (counter size n=10 and clock frequency $F_{clk} = 12MHz$). The dynamical response of both division and square root circuit depends on three parameters:

- Digital clock frequency F_{clk}.
- Counter size n.
- p_2 or y_2 value in division and root square circuit, respectively.

Equations (4) and (5) are the time constant for division

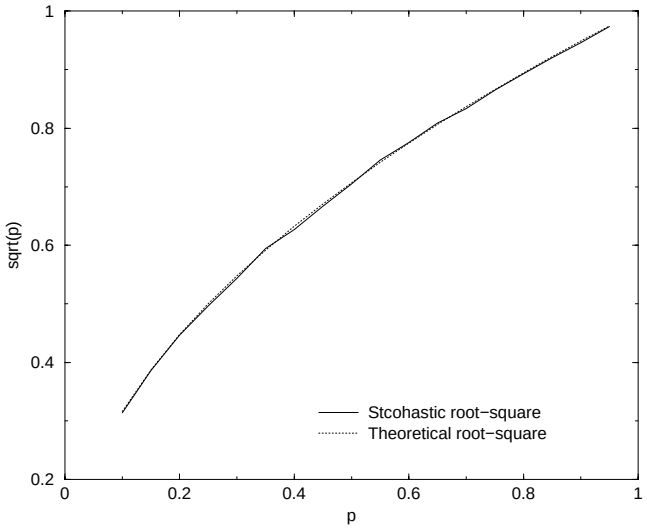

Figure 10: Experimental results of the stochastic square root circuit

and square root circuits [13]:

$$\tau_{div} = \frac{2^n}{p_2 F_{clk}} \tag{4}$$

$$\tau_{root} = \frac{2^n}{y_2 F_{clk}} \tag{5}$$

In order to obtain a reasonable speed in both circuits, the clock frequency must be much higher than the signal dynamic we want to process. This restriction is due to the fact of representing the information as a probability in a sequence of binary pulses. Nevertheless, notice that we are performing pseudo-analog operations working with the mean value of the random pulse stream. By using the conversion schemes of section II, we are able to implement a very simple and purely digital solutions to real applications, like the ones referenced in [6] and [7].

4. CONCLUSIONS

The use of stochastic logic has provided a set of very simple arithmetic circuits constituting a design library suitable to solve algebraic and differential equations. Some applications has been recently developed using stochastic arithmetic. Particularly. a set of algebraic equations has been solved in [6] and a set of differential equations has been solved in [7].

There are other digital and analog techniques to solve division and square-root operations [14]. The main advantage of stochastic approach is that can be used massively in conjunction with other stochastic circuits, carrying out a serialized information and with a very simple analog and digital interface. This approach allows for the realization of quasi-analog functions but taking advantages of digital circuits properties.

5. REFERENCES

[1] B.R. Gaines, *Stochastic computing systems*, Advances Inform. Syst. Sci., vol. 2, pp. 37-172, 1969.

[2] C.L. Janer, J.M. Quero, J.G. Ortega and L.G. Franquelo, *Fully Parallel Stochastic Computation Architecture*, IEEE Trans. Signal Processing, vol. 44, no. 8, Au. 1996, pp. 2110- 2117, Aug. 1996.

[3] M. Hennecke, *RANEXP: experimental random generator package*, Computer Physics Communications 79, pages 261-267, 1994.

[4] S.W. Golomb, *Shift Register Sequences*, Holden-Day, San Francisco, 1967.

[5] J. Tomberg and K. Kaski, "*Feasibility of Synchronous Pulse-Density Modulation Arithmetic in Integrated Circuit Implementations of Artificial Neural Networks*", Proc. of Intnl. Symp. on Circ. and Sys., ISCAS'92, vol. 5, pp. 2232-2235, San Diego, CA, May 1992.

[6] S.L. Toral, J.M. Quero and L.G. Franquelo, "*Power Energy Metering based on Random Signal Processing (EC-RPS)*", ISCAS'98, Monterrey, California, May-June 1998.

[7] J.M. Quero, S.L. Toral, J.M. Carrasco, J.G. Ortega and L.G. Franquelo, "*Continuous Time Controllers using Digital Programmable Devices*", IECON'99, San Jose, CA, Dec. 1999 (accepted for publication).

[8] J.M. Quero, C.L. Janer, J.G. Ortega and L.G. Franquelo, "*D/A Converter ASIC Uses Stochastic Logic*", EDN, pp. 86-88, Oct. 1996.

[9] J.G. Ortega, C.L. Janer, J.M. Quero, L.G. Franquelo, J. Pinilla and J. Serrano, "*Analog to Digital and Digital to Analog Conversion Based on Stochastic Logic*", IEEE Intl. Conf. on Industrial Electronics, IECON'95, Orlando, Nov. 1995.

[10] S.L. Toral, J.M. Quero, J.G. Ortega and L.G. Franquelo, "*Stochastic A/D Sigma Delta Converter on FPGA*", Intnl. Proc. of 42nd Midwest Symposium on Circuits and Systems, MWSCAS'99, Las Cruces, Nuevo Mexico, Aug. 1999.

[11] S. Ribeiro, "*Random-pulse machines*", IEEE Transactions on Electronic Computers, vol. EC-16, no. 3, pp. 261-276, June 1967.

[12] S.L. Toral, J.M. Quero and L.G. Franquelo, "*Reactive Power and Energy Meter based on Random Signal Processing*", DCIS'98, Madrid, pp. 346-351, Nov. 1998.

[13] J.M. Quero, S.L. Toral, J.G. Ortega and L.G. Franquelo, "*Continuous Time Filter Design Using Stochastic Logic*", Intnl. Proc. of 42nd Midwest Symposium on Circuits and Systems, MWSCAS'99, Las Cruces, Nuevo Mexico, Aug. 1999.

[14] M.D. Ercegovac and T. Lang, "*Division and Square-Root: Digit-Recurrence Algorithms and Implementations*", Boston, Kluwer Academic, 1994.